



# **5231 - APLICACIÓ PER A LA PREDICCIÓ AUTOMÀTICA DE LA DATA D'ADQUISICIÓ D'UNA IMATGE**

Memòria del Projecte Fi de Carrera  
d'Enginyeria en Informàtica  
realitzat per  
Eloi Berenguer Arroyo  
i dirigit per  
Robert Benavente Vidal  
Bellaterra, 19 de Juny de 2013



El sotasignat, Robert Benavente Vidal

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

**CERTIFICA:**

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en

I per tal que consti firma la present.

Signat: .....

Bellaterra, 19 de Juny de 2013

# Índex

<b>Capítol 1: Introducció</b>	<b>5</b>
1.1. Objectius del projecte.....	5
1.2. Motivacions.....	6
1.3. Estat de l'art.....	7
1.4. Estudi de viabilitat del projecte.....	8
1.5. Planificació temporal del treball.....	9
1.6. Organització de la memòria.....	10
<b>Capítol 2: Fonaments Teòrics</b>	<b>11</b>
2.1. Visió per computador.....	11
2.2. Classificadors.....	12
2.2.1. Support Vector Machine.....	13
2.3. Imatges.....	17
2.4. Color.....	18
2.4.1. RGB.....	19
2.4.2. CIELAB.....	20
2.4.3. HSV.....	21
<b>Capítol 3: Dating Historical Color Images</b>	<b>22</b>
3.1. Introducció.....	22
3.2. Descriptos.....	22
3.2.1. Color de co-currència d'histograma.....	23
3.2.2. Probabilitat condicional de saturació donat un to.....	23
3.2.3. Histograma del to.....	24
3.2.4. Descriptor gist.....	25
3.2.5. Petites imatges.....	25
3.2.6. Histograma del color $L^*a^*b$ .....	26
3.3. Support Vector Machine.....	26
3.4. Experiments.....	28

<b>Capítol 4: El nou descriptors</b>	<b>31</b>
4.1. Color Naming.....	31
4.2. Procés d'adaptació del Color Naming.....	32
4.3. Resultats.....	32
<b>Capítol 5: Interfície gràfica per classificar imatges</b>	<b>34</b>
<b>Capítol 6: Conclusions i possibles millores</b>	<b>37</b>
6.1. Conclusions.....	37
6.2. Possibles millores.....	37
<b>Bibliografia</b>	<b>39</b>

# Capítol 1:

## Introducció

### 1.1.Objectius del projecte

En una societat on cada cop més la majoria de dades s'estan passant a format digital, comencen a aparèixer grans volums de dades que necessiten ser classificats per a que sigui més fàcil el seu ús. Les imatges per exemple són un d'aquests grans volums de dades que es poden classificar segons una gran varietat de característiques, per exemple, pels colors que conté, data en què es va adquirir, objectes que hi apareixen... Apart s'ha de tenir en compte, el gran creixement que ha sofert el món de les imatges, gràcies a l'expansió de les càmeres digitals, l'aparició d'una gran quantitat de programes utilitzats per el retoc i modificació de les imatges i l'aparició de xarxes socials com Facebook, Flickr, Instagram on es poden compartir gran quantitat d'imatges. Per tant sembla ser que cada cop més la classificació d'imatges segons un criteri serà més necessària. Però per una sola persona, encara que hagi estat entrenada en la classificació, se li fa molt difícil i dura la correcta classificació d'una gran base de dades d'imatges. Apart de la gran quantitat de temps que li reportaria a una persona també s'ha de tenir present que la persona encarregada seguint un criteri més subjectiu podria errar més fàcilment que una computadora.

En aquest projecte tractarem el problema de la creació d'una aplicació que permeti als usuaris descobrir en quina dècada entre els anys 30 i els anys 70 va ser realitzada una fotografia. Creiem que aquesta aplicació podria ajudar a bastant gent ja fós en el seu lleure o en món professional. Per exemple, als historiadors i periodistes els podria ajudar a l'hora de buscar una fotografia d'una època que els interessés. L'usuari que utilitzi l'aplicació tindrà la possibilitat de carregar una imatge en l'aplicació. Un cop l'imatge està carregada l'usuari podrà escollir un descriptor que permet a l'aplicació extreure una sèrie d'informació de la imatge. Un cop l'aplicació aconsegueix la informació i amb l'ajuda del classificador Support Vector Machine l'aplicació decidirà a quina dècada pertany la imatge. Per realitzar aquesta funcionalitat utilitzarem el mètode proposat a l'article acadèmic “Dating Historical Color Images” [1], el qual ens detalla els passos realitzats per tal d'aconseguir la datació de imatges segons certs aspectes d'aquestes. Però els resultats que van aconseguir tot i ser bastant positius no deixaven de tenir

un índex d'encert relativament en la classificació d'imatges. Per tant és va arribar a la conclusió que el nostre objectiu principal seria la creació d'una aplicació que presentes millors resultats que els de l'article[1]. D'aquest objectiu principal s'en deriven els següents subobjectius:

- La comprovació dels resultats obtinguts per l'article [1].
- L'estudi d'altres descriptors i La millora dels resultats del article [1], utilitzant un nou descriptor.
- La creació d'una interfície còmoda i fàcil d'utilitzar que permeti carregar imatges que conté un ordinador i que es visualitzin dintre la interfície, i aplicar el mètode utilitzat en l'article a les imatges carregades, per tal de classificar correctament les imatges.

## 1.2.Motivacions

Les motivacions que em van portar a escollir aquest projecte van ser dos: el fet de tractar el tema de les fotografies i el de que es tractés d'un projecte que estès en el camp de la visió per computador.

El fet que el projecte tractés sobre la classificació d'imatges em va semblar molt interessant, ja que des de petit sempre m'ha agradat molt la fotografia i sobretot tafanejar les fotografies velles que hi havien per casa. Per això vaig creure que crear una aplicació que ajudés en la classificació d'imatges podria ser molt interessant, bàsicament pel fet que molts cops es tenen moltes fotografies velles per casa i no se sap quan van estar realitzades i crec que aquesta aplicació pot ajudar en aquesta tasca.

El fet que el projecte estès dintre el camp de la visió per computador em va motivar degut que l'any passat l'assignatura de visió per computador em va cridar molt l'atenció. Apart actualment la visió per computador és un camp que esta creixent molt ràpid i és molt interessant, ja que permet la realització de projectes que podrien ajudar a la societat d'una manera molt gran.

## 1.3.Estat de l'art

Aquest projecte es molt interessant ja que el tema de la datació d'imatges no ha estat gaire desenvolupat per no dir gens. Ja que hem estat buscant aplicacions comercials que duguessin a terme la datació d'imatges i no n'hem trobat cap. L'únic que hem trobat ha estat el projecte **Dating Historical Color Images** [1]. Que és un projecte que s'esta duent a terme conjuntament en la Brown University i la Carnegie Mellon University (Estats Units) basat en la datació d'imatges. Per el nostre projecte utilitzarem part del codi realitzat en aquest projecte. El que si que hem trobat han estat altres aplicacions o projectes de caire acadèmic per classificar o modificar imatges en altres aspectes, a continuació en destacarem alguns:

- **Photo and Video Quality Evaluation: Focusing on the Subject** [2] és un Projecte de la The Chinese University of Hong Kong, que pretén qualificar la qualitat de la fotografia depenent de si el que l'ha realitzat és un professional o un aficionat. Ja que per un ésser humà es molt fàcil identificar si la fotografia es bona o no, però en canvi per una màquina és molt difícil. Per tal de que una màquina pugui qualificar la qualitat d'una imatge es centren en certs aspectes d'aquesta com per exemple la composició, la llum, etc. I a partir d'aquests apliquen un classificador per determinar la qualitat de la imatge.
- **What makes an image memorable?**[3] És un projecte del Massachusetts Institute of Technology en el qual realitzen un estudi sobre els elements que han de formar part d'una imatge per tal que aquesta sigui fàcil de recordar.
- **Color Harmonization** [4] projecte de la Tel Aviv University en que el seu objectiu es crear una aplicació que modifiqui alguns colors de les imatges per tal que tots els colors que apareixen en la imatge siguin harmònics entre ells.

## 1.4. Estudi de viabilitat del projecte

Per tal que el funcionament de l'aplicació tingui els millors resultats en termes temporals es recomana que el ordinador en que volem utilitzar l'aplicació tingui com a mínim una CPU Intel Core i7-2600K i una targeta gràfica relativament bona (NVIDIA, AMD...). Encara que si l'ordinador té una CPU o una targeta gràfica pitjor no es cap problema l'únic que possiblement l'aplicació pot tardar una mica més al realitzar les seves funcionalitats.

El llenguatge de programació que em escollit per tal de crear el projecte serà amb el Matlab, ja que don molta facilitat a l'hora de tractar amb imatges. En primer moment l'aplicació s'executara en el sistema operatiu Windows, ja que té una quantitat més elevada d'usuaris. A continuació s'exposen els requeriments funcionals i no funcionals que ha de tenir l'aplicació.

### **Requeriments funcionals:**

1. Interfície còmode i fàcil de utilitzar que permeti carregar imatges que conté un ordinador i que es visualitzin dintre la interfície.
2. Detecció de a quina dècada pertany la imatge carregada i presentar els resultats per pantalla.
3. Deixar escollir a l'usuari quin descriptor vol utilitzar per a detectar a quina dècada pertany l'imatge

### **Requeriments no funcionals:**

1. Estabilitat de l'aplicació, efectuant control d'errors i tractament d'excepcions. Que l'aplicació no es quedi bloquejada.
2. Rendiment temporal acceptable, que l'espera de detecció de quina dècada pertany la imatge o la de modificació no duri més de uns 10 segons.
3. Aplicació de caràcter gratuït.

Creiem que el projecte es viable, ja que en temes de H/W i software no és molt exigent apart els requeriments funcionals i no funcionals creiem que es podran assolir sense cap problema. Apart que en el mercat no hi ha cap aplicació d'aquest tipus i pot ser bastant interessant la seva creació.



## 1.5. Planificació temporal del treball

Les jornades de treball es consideren que son de mitjana de 2 hores. Per tant si el projecte dura en total 126 dies i tenint en compte que la memòria es realitzara durant la realització de les altres tasques, tenim que en total dedicarem mes de 476 hores en el projecte. La següent imatge mostra la planificació proposada al principi del projecte:

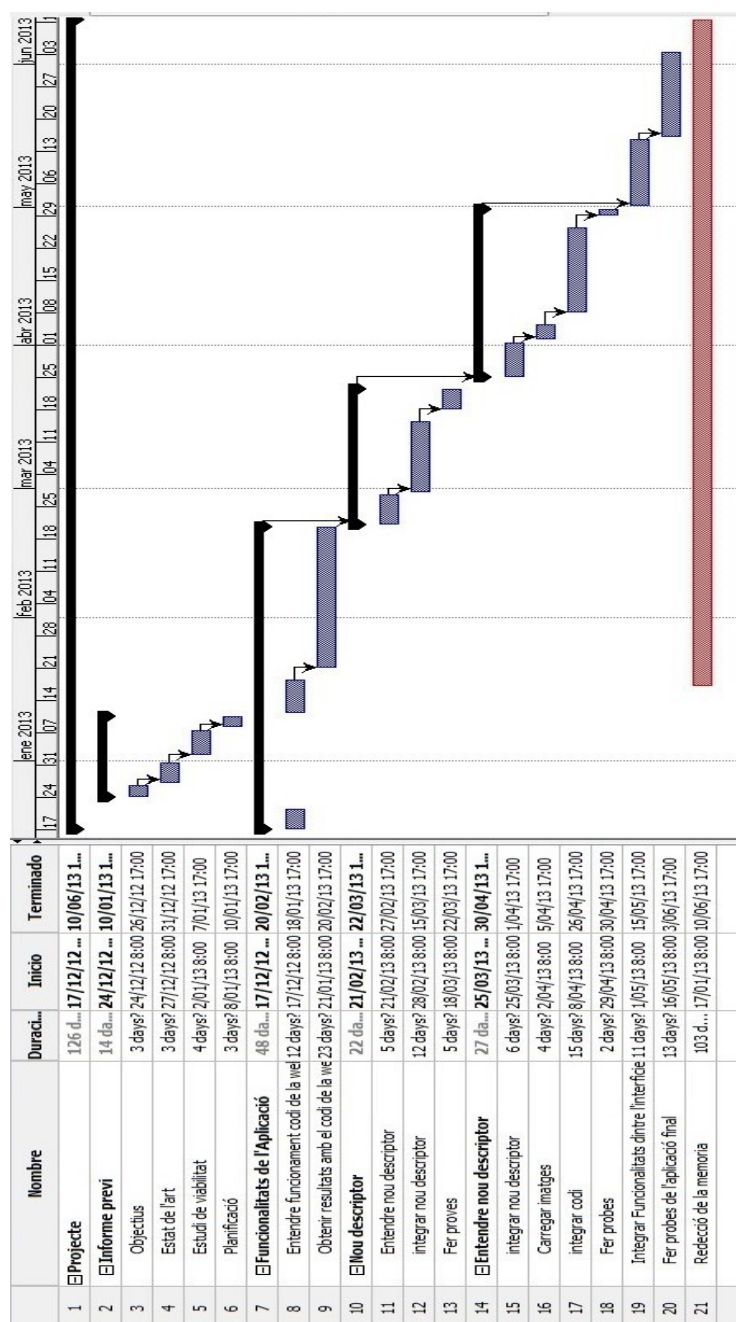


Figura 1: Planificació pensada al principi del projecte

## 1.6.Organització de la memòria

L'organització de la memòria està feta de tal manera que representa l'evolució que ha seguit el projecte i es divideix en 6 capítols.

El primer capítol, es donen a conèixer els objectis i les motivacions del projecte, juntament amb l'estat d'art.

En el capítol 2 es comenten de forma introductòria els fonaments teòrics que fan possible aquest projecte, com la visió per computador, que és un classificador i quins diferents tipus n'existeixen, un breu resum de les diferents maneres en que es poden representar en color en les imatges i una explicació dels aspectes que poden ajudar a classificar degudament una imatge.

El capítol 3 Primerament es comenta l'article utilitzat i en que consistia. Després s'explica exhaustivament com funcionen cada un dels descriptors utilitzats i el funcionament de l'algorisme de classificació utilitzat. Ja per acabar al punt 3 es comenten els resultats obtinguts en els experiments realitzats en l'article i els obtinguts ens els nostres experiments.

En el capítol 4 es comenta com funciona el nou descriptor que em adaptat per l'aplicació. Apart també es comenta com l'hem adaptat i quins son els resultats que hem obtingut amb la seva execució.

En el capítol 5 es parla de com i quins criteris hem seguit per tal de crear la interfície que utilitza l'aplicació per tal de classificar les imatges.

En el capítol 6, finalment, es presenten les conclusions obtingudes del projecte i es comenten possibles millores que es podrien realitzar al projecte per tal de millorar-lo i fer-lo més atractiu.

## Capítol 2:

# Fonaments teòrics

## 2.1. Visió per computador

La visió per computador és un camp de l'intel·ligent artificial que inclou mètodes per adquirir, processar, analitzar i comprendre les imatges i, en general, les dades d'alta dimensió de la vida real per tal de produir informació numèrica o simbòlica. Per tal d'interpretar imatges la visió per computador es val de l'extracció de descriptors, cal tenir present que les imatges sempre es poden representar mitjançant estructures més abstractes i senzilles [5][6].

La visió per computador és un camp que té relació amb moltes més àrees. Segurament amb la que més se la relaciona es amb la intel·ligència artificial, ja que les dos àrees comparteixen temes com reconeixement de patrons i tècniques d'aprenentatge. Per tant, la visió per ordinador es veu de vegades com una part de la intel·ligència artificial. Una altre camp que es podria relacionar amb la visió per computador seria la física, sobretot per el fet dels sensors i de les òptiques. En la neurobiologia també hi trobem una relació ja que actualment un sub-camp de la visió per computador s'encarrega d'imitar el processament i el comportament de sistemes biològics, a més alguns mètodes basats en l'aprenentatge desenvolupades en la visió per computador (per exemple, les en xarxes neuronals) tenen el seu origen en la biologia. Altres camps amb que té relació serien les matemàtiques, el processament de senyals, control automàtic... Però segurament amb el camp que hi està més relacionat es amb el processament d'imatges, anàlisi d'imatge i la visió de la màquina. Hi ha un solapament significatiu en la gamma de tècniques i aplicacions que aquestes cobreixen. Això implica que les tècniques bàsiques que s'utilitzen i es van desenvolupar en aquests camps són més o menys idèntics, cosa que pot ser interpretat com que només hi ha un camp amb diferents noms.[6]

Dintre de la visió per computador hi ha 3 nivells de jerarquia, que depenen de l'ús que és realitzi [5][7]:

- Nivell baix: bàsicament es basa en el processament d'imatges (captura, digitalització, emfatitzar...)

- Nivell mitjà: extracció de descriptors no trivials però senzills, reconeixement d'objectes i la interpretació d'escenes 3D.
- Nivell alt: Obtenció de descriptors complexos que permeten la interpretació d'una escena. Utilització de coneixements i aprenentatge d'informació

Actualment la visió per computador és una de les disciplines informàtiques que està experimentant un gran increment. Això es degut simplement a que cada cop se l'hi estan trobant més utilitats i la visió per computador està evolucionant a una velocitat molt elevada. Alguns exemple podrien ser la detecció de cares en sistemes de seguretat, o l'automatització de la conducció de vehicles... Però bàsicament els seus objectius son els següents:

- Obtenció i millora ( reducció de soroll, enriquiment del detall) d'imatges.
- Detecció, localització o reconeixement d'objectes en una imatge.
- Obtenció de descriptors d'una imatge per poder-la interpretar.
- Interpretació d'una escena ja sigui en temps real.

## 2.2.Classificadors

En intel·ligència artificial la classificació és el problema d'identificar a quina categoria d'un conjunt de categories pertany una nova mostra. Els que solucionen aquest problema són els classificadors. Bàsicament l'aplicació dels classificadors es poden dividir en dues fases, la fase d'aprenentatge i la fase de test. Els classificadors aconseguixen classificar correctament una nova mostra a partir de la construcció de models d'aprenentatge. Els models són espais on les diferents classes en què es volen classificar les dades estan separades per un cert espai o segueixen un mateix patró. En la fase d'aprenentatge es quan és creen els models que ajudaran a classificar correctament una mostra. En canvi la fase de test consisteix en introduir una nova mostra dintre el model d'aprenentatge, de tal manera que la seva posició dintre el model d'aprenentatge ens indicarà a quina classe correspon la mostra. Els classificadors es poden dividir en varis grups segons el tipus d'aprenentatge que utilitzen, encara que els dos més importants són els que utilitzen aprenentatge no supervisat i els que utilitzen aprenentatge supervisat.

En el cas dels algoritmes d'aprenentatge no supervisat l'algorisme disposa d'un conjunt d'exemples, però no de la classe que els hi correspon, això és degut a que els exemples no estan etiquetats. Per tant l'algorisme ha de ser capaç de trobar patrons similars entre els exemple per tal de poder crear correctament les classes. Els algorismes de categorització (en anglès, *clustering*) són un exemple d'aquests tipus d'aprenentatge [8].

En canvi, en els algoritmes d'aprenentatge supervisat, el conjunt d'exemple sí que tenen indicada a quina classe corresponen. En aquest cas l'objectiu de l'entrenament consisteix, a partir dels exemples obtinguts al principi, deduir una funció que sigui capaç de preveure a quin conjunt correspon una mostra d'entrada, després d'haver vist les dades d'entrenament. En el nostre projecte utilitzarem el support vector machine (SVM), que és un algoritme d'aprenentatge supervisat. En el següent apartat s'expliquen els diferents tipus de SVM que hi han i com funcionen.

### **2.2.1.Support Vector Machine**

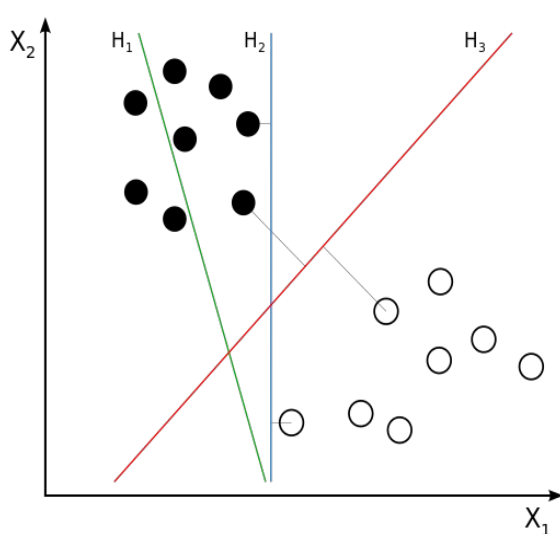
L'inici de la teoria en la que es basen els classificadors SVM data dels anys setanta amb els treballs de Vapnik [9]. En els anys noranta el mètode va ser generalitzat, en l'actualitat presenta un gran interès. Les SVM són noves estructures d'aprenentatge basades en la teoria estadística de l'aprenentatge. Es basen en transformar l'espai d'entrada en un altre de dimensió superior (infinita) en el que el problema pot ser resolt mitjançant un hiperplà òptim (de màxim marge). Les SVM presenten un bon rendiment al generalitzar en problemes de classificació, encara que no incorpora coneixement específic sobre el domini. Per tant la solució no depèn de l'estructura del plantejament del problema. La idea és construir una funció classificadora que minimitzi l'error en la separació dels objectes donats (error en classificació) i maximitzi el marge de separació (millora la generalització del classificador).

## SVM lineals per a dos classes

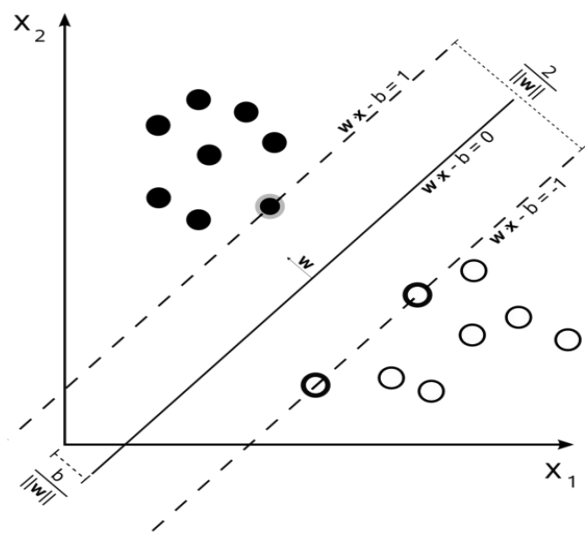
Donat un conjunt de punts, subconjunt d'un conjunt major (espai), en el que cada un d'ells pertany a una de les dos categories possibles, un algoritme basat en SVM construeix un model capaç de predir si un punt nou (desconeixem la seva categoria) pertany a una categoria o a una altra. Com en la majoria dels mètodes de classificació supervisada, les dades d'entrada son vistos com un vector p-dimensional (una llista de p números). En l'aprenentatge el SVM busca un hiperplà que separi de forma òptima els punts d'una classe dels de l'altre.

En la cerca de l'hiperplà que separi les dues classes és on resideix la característica fonamental de les SVM, ja que aquest tipus d'algoritmes busquen l'hiperplà que tingui la màxima distància (marge) amb els punts que estan més aprop d'ell mateix, en la figura 2 és pot observar amb més detall. Per això també a vegades a les SVM se les coneix com classificadors de marge màxim. D'aquesta forma, els punts que són etiquetats com una categoria estaran a una costat de l'hiperplà i els casos que es troben en l'altre costat pertanyeran a una altre categoria.

S'anomena atribut, a la variable predictora i característica, a un atribut que és usat per definir l'hiperplà. L'elecció de la representació més adequada de l'univers a estudiar, es realitza mitjançant el procés denominat selecció de característiques. Els vectors formats pels punts més propers a l'hiperplà se'ls anomena vectors de suport.



**Figura 2:** El hiperplà H1 no separa les classes. El hiperplà H2 sí que les separa, però amb un petit marge, mentre que l'hiperplà H3 les separa amb el seu marge màxim.



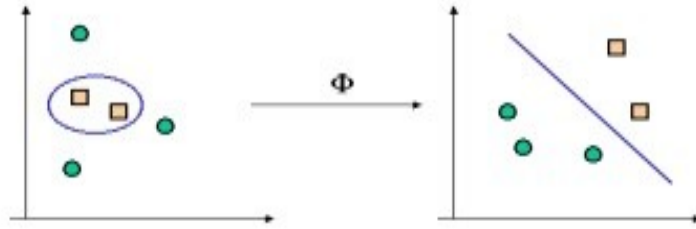
**Figura 3:** Correcte divisió entre dos classes, on s'observen els marges.

Si representem de manera matemàtica els paràgrafs anteriors, tindríem que si tenim un conjunt d'entrenament  $D$  i un conjunt de  $n$  punts de la forma  $D = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p, y_i \in \{-1, 1\}\}_{i=1}^n$ .  $y_i$  fa referència a quina classe pertany  $\mathbf{x}_i$ . Nosaltres volem trobar el marge màxim de l'hiperplà que divideix els punts que tenen  $y_i = 1$  i els que tenen  $y_i = -1$ . Qualsevol hiperplà pot ser escrit com el conjunt de punts  $\mathbf{x}$  que satisfà  $\mathbf{w} \cdot \mathbf{x} - b = 0$ , on  $\mathbf{w}$  denota el vector normal del hiperplà. Per tant si les dades d'entrenament son linealment separables, podem seleccionar dos hiperplans de la manera que ells separin les dades i si no hi ha punts entre ells, després provar de maximitzar la distancia. La regió entre els dos hiperplans s'anomena el marge. Els hiperplans poden ser descrits amb aquestes equacions  $\mathbf{w} \cdot \mathbf{x} - b = 1$  i  $\mathbf{w} \cdot \mathbf{x} - b = -1$ . Per tant podem dir que qualsevol punt  $\mathbf{x}_i$  que compleixi la condició  $\mathbf{w} \cdot \mathbf{x}_i - b \geq 1$  pertanyera a la primera classe, mentre que el punt  $\mathbf{x}_i$  que compleixi la condició  $\mathbf{w} \cdot \mathbf{x}_i - b \leq -1$  pertanyera a la segona classe. Aquestes dues equacions poden ser reescrites com  $y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1$ , per tot  $1 \leq i \leq n$ . [9]

Idealment, el model basat en SVM hauria de produir un hiperplà que separés completament les dades d'entrenament en dos categories. Però, una separació perfecta no sempre és possible. Per tant es va introduir una nova millora en els classificadors SVM, la qual permetia certa flexibilitat. Això ho aconseguia amb una variable d'ajustament  $C_i$  que controla la compensació entre errors d'entrenament i els marges rígids, permeten ajustar el grau de classificació errònia sobre les dades  $\mathbf{x}_i$ , creant així un marge tou que permet alguns errors en la classificació a la vegada que els penalitza. De tal manera que l'equació anterior quedaria reescrita com a  $y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 - C_i$ , per tot  $1 \leq i \leq n$ . [9]

## SVM no lineal

Hi han ocasions en que divisió entre les dues categories no és lineal. Per aquests casos es va crear el classificador SVM no lineal. El SVM lineal aconsegueix dividir les dues categories portant les dades d'entrenament a un nou espai euclidià  $H$  (possiblement de dimensió infinita), això s'aconsegueix mitjançant una transformació  $\Phi$ . Transformem l'espai d'entrada a un altre espai que és de més alta dimensionalitat en el que les dades si son separables linealment. D'aquesta forma, podem aplicar sobre aquest nou espai el procediment lineal vist en l'apartat anterior. [10]



**Figura 4:** exemple de com un espai no lineal passa a ser lineal amb una funció  $\Phi$

Aquesta operació de transformació es realitza utilitzant la separabilitat de patrons. De tal manera que l'algoritme d'entrenament només dependrà de les dades a través de productes escalars en  $H$ , és a dir, de funcions de la forma  $\Phi(x_i) \cdot \Phi(x_j)$ . Si existís una funció de nucli  $K$  tal que  $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$ , podríem utilitzar únicament  $K$  en l'algoritme d'entrenament sense tenir que conèixer explícitament  $\Phi$ . En la majoria d'ocasions  $H$  és de dimensió infinita i no seria senzill treballar explícitament amb  $\Phi$ . Però en canvi, si substituïm  $\Phi(x_i) \cdot \Phi(x_j)$  per una funció  $K(x_i, x_j)$  al llarg de tot l'algoritme d'entrenament obtindrem un SVM que estarà en un espai de dimensió infinit i que operarà pràcticament en la mateixa quantitat de temps i la separació seguirà sent lineal, però en un espai diferent.[10] Les funcions de nucli més comuns per el reconeixement i la classificació de patrons són les següents:

- $K(x, y) = (x \cdot y + 1)^p$
- $K(x, y) = \exp(-\gamma \|x - y\|^2)$ , on  $\gamma > 0$  alguns cops es parametriza usant  $\gamma = 1/2\sigma^2$
- $K(x, y) = \tanh(k \cdot x \cdot y + c)$ , on  $k > 0$  i  $c < 0$

## SVM multi-classe

Fins ara els SVM comentats eren utilitzats només per classificacions binaries, però en molts casos necessitem fer classificacions on hi hagin més de 2 categories. Per aquests casos es va crear el SVM multi-classe, que és un classificador que ens permet fer classificacions per més de dos categories.



Per tant considerem el cas en què el conjunt d'etiquetes possibles és  $\{\theta_1, \theta_2, \dots, \theta_L\}$ , sent  $L > 2$  i sense una relació d'ordre definida en l'etiquetatge. Sigui  $Z$  el conjunt d'entrenament definit per  $Z = \{(x_1, y_1), \dots, (x_n, y_n)\}$ , es construeixen els subconjunt  $Z_k = \{(x_i, y_i), \text{ tals que } y_i = \theta_k\}$  que determinen una partició  $Z$ . La forma més habitual com s'utilitzen les SVM per resoldre problemes de multi classificació admet dos tipus d'arquitectura:

- Màquines biclassificadores SVM generalitzades: Construeixen una funció classificadora global a partir d'un conjunt de funcions classificadores dicotòmiques (cada una de les biparticions que componen l'arquitectura de la SVM multi-classe i son capaces de generar una predicció o resposta parcial). Utilitzen la Descomposició-Reconstrucció, primerament s'utilitza un esquema de descomposició que transforma les  $L$  particions en una sèrie de  $L$  biparticions,  $\theta_1, \dots, \theta_L$ , posteriorment un mètode de reconstrucció realitza la fusió de les prediccions dels  $L$  classificadors per seleccionar una de les  $K$  classes com a resposta final.
- Màquines multi-classificadores SVM: Construeixen una funció classificadora global directament, considerant totes les diferents classes a la vegada.

Dintre de l'esquema de descomposició, les màquines més utilitzades son:

- Màquines 1-vs-r SV (one-versus-rest): on cada funció classificadora parcial  $f_i$ , enfronta a la classe  $\theta_i$  contra la resta de classes. Es requereixen  $L=k$  biclassificadors, on  $k$  és el numero de classes que es tenen.
- Màquina 1-vs-1 SV (one-versus-one): on cada funció classificadora parcial  $f_{ij}$ , enfronta la classe  $\theta_i$  contra la classe  $\theta_j$ , sense considerar les classes restants. Es requereixen  $L=k \cdot (k-1)/2$  biclassificadors.

## 2.3.Imatges

Les fotografies són un dels elements més importants del projecte. Ja que són l'element a classificar i el que ens aporta la informació que necessitem per tal d'aconseguir-ho. Els humans, sempre hem volgut immortalitzar cada cosa que ens envolta i els moments importants en les nostres vides, però no va ser fins al 1816 que varem començar a realitzar les primeres fotografies. En el moment que les càmeres van estar a l'abast del gran públic, l'aparició de

fotografies va créixer exponencialment de tal manera que actualment hi ha un gran volum d'informació fotogràfica arreu del món. L'empresa Flickr juntament amb una sèrie de bancs de fotografies han optat per digitalitzar aquestes fotografies per tal de que no es perdin i nosaltres ens em valgut d'aquesta base de dades per tal de crear el nostre projecte.

Des de que s'han tingut grans quantitats de fotografies s'ha intentat tenir-les ben classificades ja fos per èpoques o llocs o persones que hi apareixien, per tal de ajudar a la gent a classificar fotografies s'ha arribat al punt que han aparegut llibres que ens ajuden a classificar fotografies. Però aquests llibres de classificació estan enfocats perquè les fotografies les classifiqui una persona, ja que en alguns casos per classificar es basen amb la textura del material on estan les fotografies, el estil dels pentinats o la roba de les persones que hi surten o els objectes que hi apareixen. I actualment encara es molt difícil per la visió per computador detectar i classificar correctament segons els objectes o les coses que hi apareixen. Apart també hi ha la dificultat afegida que a la majoria de fotografies no hi surt cap persona o objecte que pugui ajudar a la classificació.

Però després de veure unes quantes fotografies un s'adona que una cosa que canvia bastant entre dècades és el color de les fotografies. Això es degut a varis factors, un seria que cada dècada podria tenir una càmera que la representes, de tal manera que cada dècada tindria un estil de color depenent de la càmera en que ha estat tirada. L'altra factor seria que les fotografies que estan en paper a mesura que van passant els anys el seu color es va deteriorant, de tal manera que depenent de quina dècada van ser tirades tindran un tipus de color o un altre. Per tant classificar una sèrie de fotografies segons la dècada i utilitzant fotografies de color és el més idoni.

## 2.4.Color

La majoria de descriptors que utilitzem en aquest projecte, estan basats en els color que contenen les imatges que volem classificar. Per tant és interessant fer una breu explicació de que és el color, i dels diferents mètodes que tenim per tal de mesurar el color en una imatge.

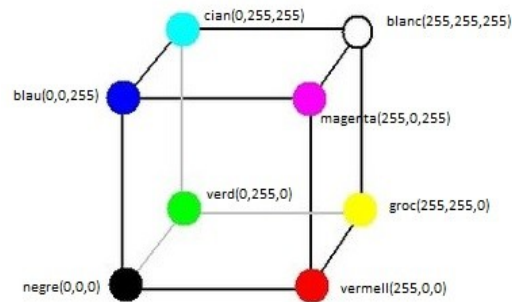
Explicat d'una manera bàsica el color és una propietat perceptiva causada per la llum quan aquesta interactua amb l'ull, el cervell i la nostra experiència. La percepció del color es veu

altament influïda pels colors adjacents en l'escena visual. Però si aprofundim més en els aspectes físics del color descobrim que el color prové de la radiació electromagnètica. La radiació electromagnètica és una mescla de radiació de diferents longituds d'ona i intensitats. Quan la radiació té una longitud d'ona compresa dins el rang visible dels humans, el qual aproximadament esta entre 380 nm i 740 nm, s'anomena llum. L'espectre de la llum emmagatzema la intensitat de cada longitud d'ona. L'espectre complet de la radiació provinent d'un objecte determina l'aparença visual d'eixe objecte, incloent-hi el color percebut. De fet es pot definir un color com el conjunt de tots els espectres que ens proporcionen la mateixa sensació de color. Per exemple una superfície que reflecteix difusament la llum de totes les longituds d'ona per igual és percebuda com a blanca, mentre que una superfície que absorbeix totes les longituds d'ona i no en reflecteix cap ho serà com a negra. Les freqüències són aproximacions i venen donades en terahertz (Thz) i les longituds d'ona, vàlides al buit, en nanòmetres(nm).

Vista la complexitat que representa la percepció del color, és molt millor definir una simplificació. Per tan es va optar per crear un mètode abstracte per especificar el color de manera matemàtica utilitzant un petit nombre de paràmetres. D'aquesta manera van començar apareix-a els models de colors que serveixen per simplificar la percepció dels colors per els humans i les computadores. D'aquesta manera amb un model de color el color es pot representar com a un conjunts de números, normalment tres o quatre valors. Els models de color més importants i utilitzats en la comuptació son: el model de color RGB, el model CIE(LAB) i el model HSV.

### **2.4.1. RGB**

El model de color RGB (Red, Green, Blue) és el més conegut per tothom ja que és el model utilitzat en els monitors i les televisions. El model de color RGB es un model de color basat en la síntesi additiva, en el que es representen els colors mitjançant els tres colors primaris (verd, vermell i blau) amb un valor per cadascun d'ells d'entre 0 i 255, on 0 es l'absència de color i 255 es el color amb màxima intensitat. De tal manera que realitzant combinacions entre els tres paràmetres aconseguim tots els colors representables. El conjunt de tots els colors es pot representar en forma de cub; cada color és un punt de la superfície o de l'interior d'aquest. L'escala de grisos és la diagonal que uneix el blanc i el negre.

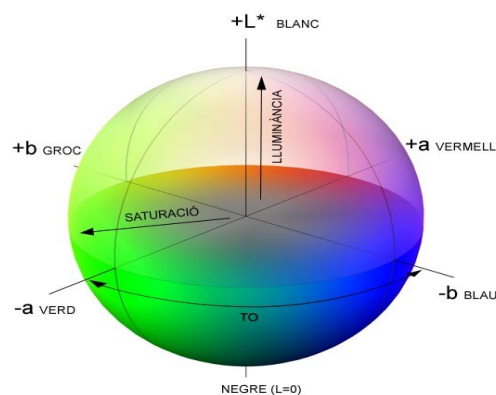


**Figura 5:** Representació del model de color RGB

## 2.4.2.CIELAB

Els models de color CIE (Commission Internationale de l'Éclairage) van ser creats per una comissió per caracteritzar racionalment els colors de llums tal com els veu el cervell humà. Per aconseguir-ho van dur a terme experiències sistemàtiques de comparació de colors per diversos observadors per poder definir un observador mitjà. En aquest model els colors no es representen per una figura geomètrica, sinó a través de coordenades i separa la luminància (intensitat) de la crominància (color independentment de la intensitat). El model més utilitzat dintre la família CIE és el CIELAB el qual separa la lluminositat del color i és el sistema de color, que assegura la coherència dels colors independentment del dispositiu. Es representa:

- El component  $L^*$  és la lluminositat, que va de 0 (negre) a 100 (blanc).
- El component  $a^*$  representa la gamma d'eixos vermell (valor positiu) → verd (negatiu) passant pel blanc (0) si la lluminositat val 100.
- EL component  $b^*$  representa la gamma d'eixos groc (valor positiu) → blau (negatiu) passant pel blanc (0) si la lluminositat val 100.

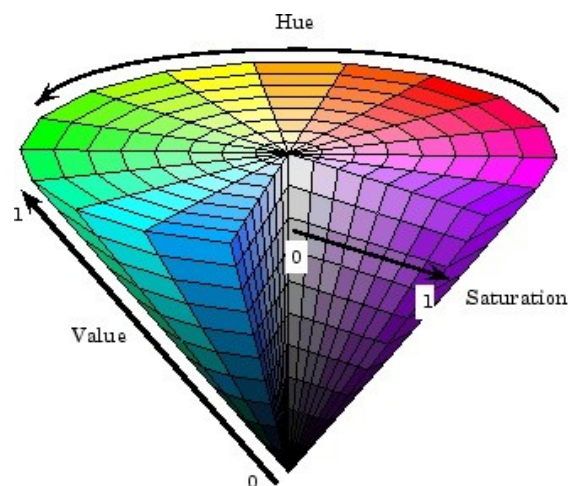


**Figura 6:** Representació del model de color RGB

### 2.4.3.HSV

El model de color HSV (Hue, Saturation, Value) o HSB (Hue, Saturation, Brightness ), el qual representa els seus components amb coordenades cilíndriques. És molt utilitzat en el món de la creació del disseny d'aplicacions i en el disseny gràfic. Les seves coordenades cilíndriques es representen:

- La tonalitat: el tipus de color, es representa com un grau d'angle els valors possibles del qual van de 0 a 360° (encara que per a algunes aplicacions es normalitzen del 0 al 100). Cada valor correspon a un color.
- La saturació: Es representa com la distància a l'eix de brillantor negra-blanca. Els valors possibles van del 0 al 100. Com menor sigui la saturació d'un color, major tonalitat grisenca hi haurà i més descolorit estarà. Per això és útil definir la insaturació com la inversa qualitativa de la saturació.
- El Valor: es la brillantor del color. Representa l'altura a l'eix blanc-negre. els valors possibles van del 0 al 100. 0 sempre és negre. Depenent de la saturació, 100 podria ésser blanc o un color més o menys saturat.



**Figura 7:** Representació del model de color HSV

## Capítol 3:

# Datació d'imatges històriques en color

### 3.1.Introducció

L'article que hem utilitzat en aquest projecte s'anomena Dating Historical Color Images i és un article conjunt de la Carnegie Mellon University, Pittsburgh i la Brown University, Providence, escrit per Frank Palermo, James Hays, and Alexei A. Efros. L'article esmentat tracta sobre la classificació d'imatges en color segons la dècada en que va ser adquirida la fotografia. Apart de l'article hem utilitzat la base de dades i el codi emprat pels autors en l'obtenció de resultats. A continuació s'expondran les parts més interessants d'aquest article.

Per tal de dur a terme el projecte hem utilitzat les eines donades en el article. Les quals han estat una sèrie de descriptors, utilitzats per aconseguir una sèrie de característiques de les imatges, i el classificador SVM. En els següents apartats explicarem més extensament cada un dels descriptors.

### 3.2.Descriptors

En visió per computador un descriptor fa referència a una quantitat d'informació, obtinguda normalment d'una imatge, que és rellevant per solucionar un problema computacional relacionat amb alguna aplicació. Per tant en el nostre cas utilitzem els descriptors sobre les diferents imatges per tal d'aconseguir de cada fotografia la suficient informació per classificar-la correctament. Els descriptors utilitzats són:

- el histograma de Co-ocurrència de Color.
- Probabilitat condicional de la saturació donat el to (hue).
- histograma del to (hue).
- Gist

- imatges de baixa resolució (tiny images).
- Histograma del color  $L^*a^*b^*$ .

En els següents subapartats expliquem els descriptors utilitzats, els dos primers son de domini més específic, mentre que els tres restants son estàndards de recuperació d'imatges o categorització d'escenes.

### 3.2.1.histograma de Co-ocurrència de Color

El descriptor histograma de Co-ocurrència de Color avalua la co-ocurrència d'intensitat dels píxels que es troben dintre un bloc de  $32 \times 32$  píxels en el centre de l'imatge. Per cada píxel que hi ha dintre el bloc  $32 \times 32$  es mira la relació que té amb els seus 4 píxels veïns. En primer moment es tenen 3 matrius de  $256 \times 256$ , cada matriu fa referència a un canal del model de color RGB. En cada matriu, el valor a la fila 'i' i a la columna 'j' correspon al número de vegades que un píxel amb un valor 'i', té un veí amb el valor 'j'. Per omplir les matrius el que es fa per exemple és obtenir el valor del paràmetre R del píxel escollit i el d'un seu veí, llavors en la posició de la matriu del descriptor on coincideixen aquests dos valors s'hi suma un 1, i així successivament per tots els veïns del píxel seleccionat i per cada un dels píxels del bloc de  $32 \times 32$ . Un cop s'han recorregut tots els píxels es passa a reduir la mida de les 3 matrius a un mida de  $32 \times 32$ , i al final es concatenen totes les matrius creant un vector de 3072 posicions.

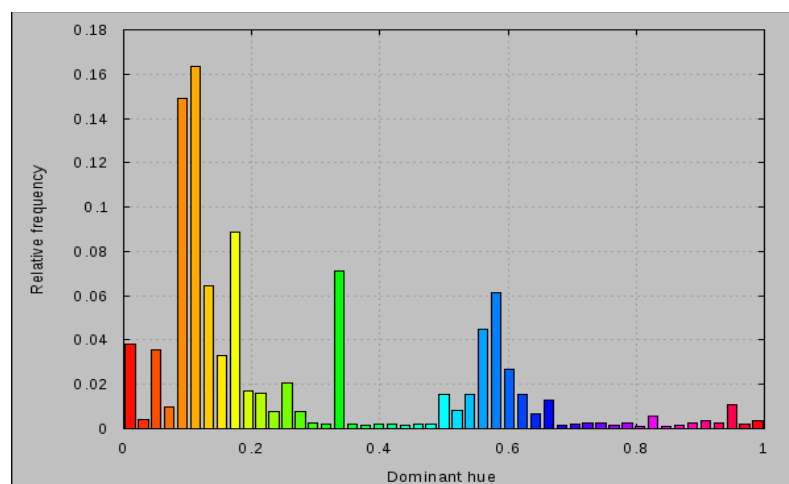
### 3.2.2.Probabilitat condicional de la saturació donat el to (hue)

L'objectiu del descriptor Probabilitat condicional de la saturació donat el to és codificar la correlació entre el to i la saturació en una imatge en l'espai CIELAB. Aquest descriptor ha estat elegit perquè els únics aspectes d'una imatge que varien segons els processos històrics en la formació d'imatges en color són certs tons i la saturació. Per tal de poder obtenir el descriptor, primerament s'han de descartar els píxels que estan massa aprop dels grisos (saturació baixa), per tal de donar mesures fiables de tonalitat i després poder estimar la distribució condicional de la saturació del píxel. El procés per obtenir el descriptor es basa en dividir tot l'espai del to

en 256 intervals, un cop tenim l'interval es calcula el nombre de píxels que estan dintre de cada interval del to. Després per cada to es divideix l'interval en dos parts, separant els que tenen més saturació dels que tenen menys saturació, i es calculen dos probabilitats, una calcula la probabilitat que el to estigui saturat i l'altre calcula la probabilitat que el to no estigui saturat. Bàsicament és fer la divisió entre els píxels que estan en una divisió segons la saturació amb els total de píxels que hi han en el to. Per tant, al final el nostre descriptor serà un vector de mida 512.

### 3.2.3.Histograma del to (hue)

El descriptor histograma del to utilitza el model de color CIELAB i es basa en la creació d'un histograma on només es presta atenció al paràmetre del to que contenen cada un dels píxels d'una imatge. En aquest cas el paràmetre del to es dividit en 128 intervals per tal de ser calculat el descriptor. Per tant cada interval del descriptor contindrà el nombre de píxels que tenen una tonalitat. L'idea d'aquest descriptor és que pugui indicar els canvis de color en els objectes o les estructures construïdes per l'home al llarg de cada una de les dècades que ens interessin.

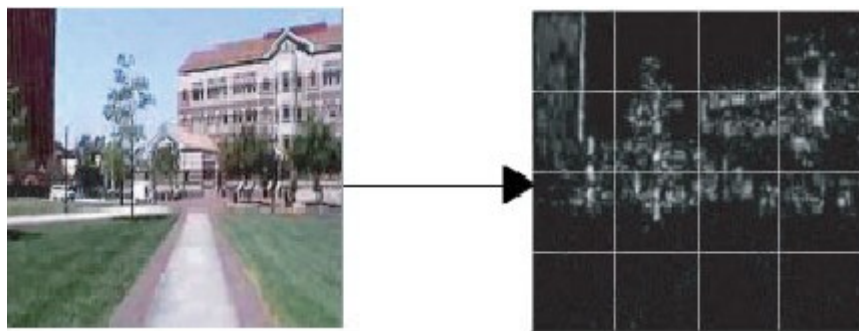


**Figura 8:** exemple de l'histograma obtingut amb el to d'una imatge



### 3.2.4.Gist

El descriptor gist s'ha demostrat que funciona bé en la classificació d'escenes i la recuperació semàntica i estructural d'escenes similars. Per tant és un descriptor que és centra en avaluar l'estructura de l'escena i aquesta característica podria servir per detectar canvis en les tendències en el tipus d'escena o la composició fotogràfica al llarg de les diferents dècades. El descriptor gist és una representació de baixa dimensió de l'estructura general de l'escena, que es calcula a partir de la resposta de la imatge a diferents filtres en diferents posicions de la imatge. En aquest article, s'utilitza una resolució espacial de 5x5, és a dir, la imatge es divideix en 5x5 regions. Per a cada regió, es calcula la mitjana de la resposta dels píxels de la regió a un conjunt de filtres amb 6 orientacions i 4 escales. Per tant, el vector del descriptor GIST té una mida de 600 posicions (tenim la resposta de les 25 regions als 24 filtres).



**Figura 9:** exemple de l'imatge obtinguda després d'aplicar el descriptor gist a una imatge

### 3.2.5.Imatges de baixa resolució (tiny images)

El descriptor imatges de baixa resolució consisteix en, donada una imatge, reduir-la a la mida de 32x32 píxels, sense importar la mida que tingués abans. Per tant, la mida d'aquest descriptor serà un vector de mida 32x32x3, el 3 que multiplica fa referència als canals RGB. Per tant aquest descriptor el que fa és que es perdi la textura de la imatge, però continua conservant el disseny general, o sigui l'estructura, i els colors predominants de l'escena.



**Figura 10:** Exemple del resultat obtingut al aplicar el descriptor imatges petites en una imatge d'entrada

### 3.2.6. Histograma del color $L^*a^*b^*$

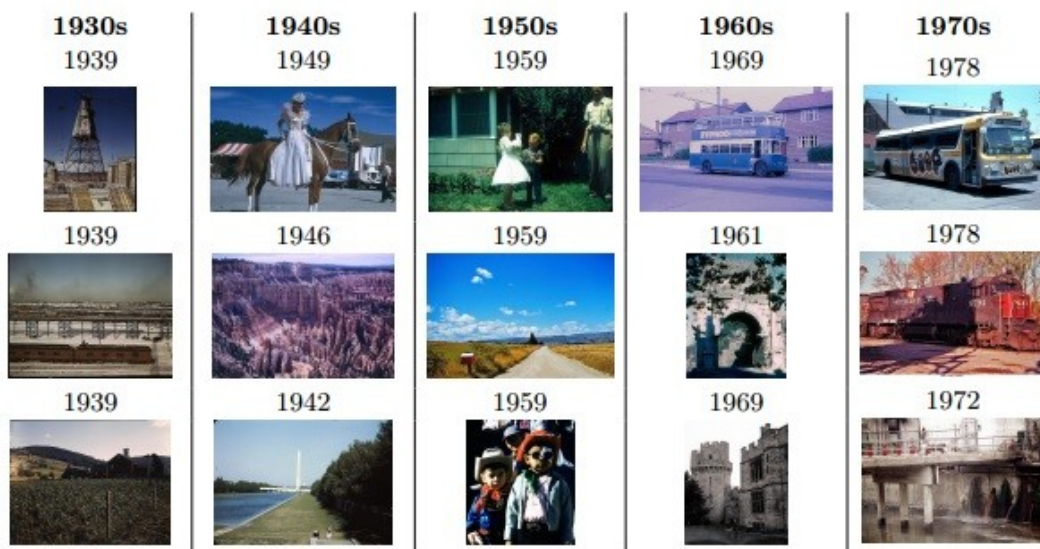
El descriptor Histograma del color  $L^*a^*b^*$  consisteix en crear una histograma d'una imatge a partir del model de color CIELAB. El model de color CIELAB és una representació del color en un sistema de 3 eixos amb tres valors que són 'L' (és la intensitat del color), 'a' (és l'eix vermell-verd) i 'b' (és l'eix groc-blau), per tal de poder crear l'histograma s'ha de dividir l'espai de color en diferents rangs. En el nostre cas el paràmetre L tindrà 4 divisions, i els paràmetres a i b en tindran 14 cadascun, fent que el nostre descriptor tingui una dimensió total de 784. L'objectiu d'aquest descriptor és indicar-nos quina quantitat de cada color hi ha en una imatge. Per tant, per cada posició en el vector que ens retorna el descriptor, ens indicarà el número de píxels que hi ha dintre d'un cert rang de 'L', 'a' i 'b' (cada rang representarà un color).

## 3.3. Support Vector Machine

Com s'especifica en l'article[1] per tal de realitzar la classificació d'imatges s'utilitza un classificador support vector machine, el qual és un classificador que utilitza un aprenentatge supervisat. Com que la base de dades d'imatges que volen classificar conté un total de cinc categories, necessitem un classificador SVM multi-classe. El tipus de multi-classe que utilitza

el SVM en l'article [1] és una màquina biclassificadora SV generalitzada amb un esquema de descomposició 1-vs-1.

La base de dades que utilitzem a l'hora de realitzar els experiments consta d'un total de 1325 imatges, les quals estan dividides en diferents subconjunts segons a quina dècada corresponen, contenint cada subconjunt un total de 265 imatges. En la figura 10 es poden observar algunes de les imatges més representatives de cada dècada. A l'hora de realitzar els experiments, la base de dades es divideix en el conjunt d'aprenentatge i el conjunt de test. El conjunt de test es crea seleccionant 50 imatges aleatòriament de cada una de les dècades. Aquest conjunt serà l'encarregat de provar el rendiment dels descriptors en el classificador SVM. El conjunt d'aprenentatge es crea amb les 215 imatges restants en la base de dades. Aquest conjunt es utilitza pel classificador SVM per crear el model d'entrenament que s'utilitzarà més tard en la fase de testeig.

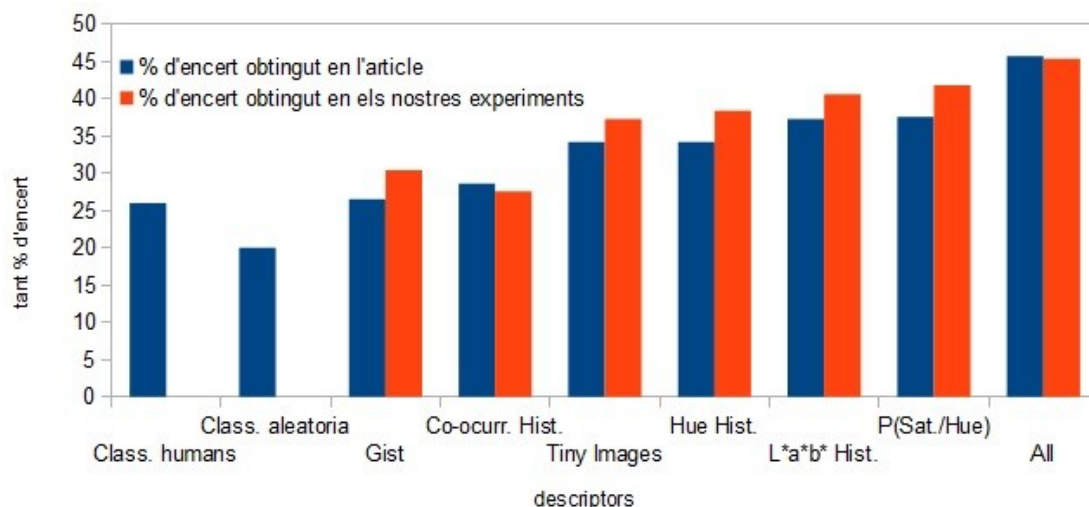


**Figura 11:** Fotografies més representatives de cada dècada dintre de la base de dades utilitzada en l'article[1]

### 3.4.Experiments

Els experiments realitzats en l'article [1] van ser executats en el sistema operatiu Linux, per tant varem creure convenient comprovar si s'obtingrien els mateixos resultats si els experiments eren realitzats en un ordinador amb el sistema operatiu Windows. De tal manera que varem utilitzar el codi proporcionat pels autors del article [1] per tal de replicar els seus experiments.

Per cada descriptor, els experiments realitzats consisteixen a crear un conjunt d'aprenentatge i un conjunt de test i avaluar el percentatge d'encert del classificador SVM. Aquesta prova es repeteix 10 vegades amb conjunts d'aprenentatge i test diferents cada vegada. De tal manera que el classificador un cop ha realitzat les 10 proves fa la mitjana entre les 10 probabilitats d'encert obtingudes en el classificador, donant-nos al final el tant per cert mitjà obtingut per el descriptor en el classificador SVM. En la figura 12 es poden veure els resultats obtinguts en cada experiment.



**Figura 12:** comparativa dels resultats obtinguts en l'article[1] i els obtinguts per els nostres experiments

Els resultats mostrats en la figura 12 ens mostren que tant els experiments realitzats en l'article i els realitzats per nosaltres no pateixen gaires variacions. Encara que s'ha de destacar que casi tots els resultats obtinguts en els nostres experiments són entre un 3% i un 4,5% millors que els resultats obtinguts en l'article. Creiem que aquesta petita millora en els nostres resultats podria ser deguda a dos factors, encara que no descartem cap altre possibilitat. Un factor podria ser l'atzar, o sigui l'atzar hauria estat el responsable de la selecció d'imatges per el test que són més

fàcils de classificar correctament i per tant fent que els nostres resultats siguin un pèl millors. Encara que és molt poc probable que aquest factor sigui el realment el causant de la millora dels resultats. L'altre factor podria ser que el codi proporcionat per els autors tingui inclosa alguna petita millora que no tenien en el codi amb el que van fer els experiments de l'article.

Si després ens fixem en els resultats individualment, o sigui observant els resultats obtinguts per cada descriptor poden treure les següents conclusions. Clarament veiem que el descriptor *gist* és el que funciona pitjor ja que només aconsegueix arribar a un 26.5% d'encert. Com diuen els autors de l'article [1], el baix nivell d'encert d'aquest descriptor ens indica que la informació sobre l'estructura de l'escena que es capta, no és tan potent com la informació de color que en aquest descriptor és ignorada. Per tant, la primera conclusió que obtindríem és que el color de la imatge és molt important per tal de poder-la classificar correctament. Per altra banda el descriptor *Tiny images*, que conté informació sobre l'estructura com el *gist*, però no ignora el color d'aquesta, funciona millor, aconseguint un 34.2% d'efectivitat. Al igual que el descriptor *Tiny images*, el *hue histogram* també té un nivell d'encert del 34.2%. Els resultats del *hue histogram*, com comenten en l'article [1], ens porten a pensar que el canvis de color que pateix l'imatge al llarg del temps poden ser un indicador útil a l'hora de classificar una imatge dintre unes certes dècades.

L'actuació del descriptor de histogrames de co-ocurrència en l'espai RGB és bastant baix aconseguint un nivell d'encert del 28.6%. Els descriptors més potents individualment són el l'histograma  $L^*a^*b^*$  i la probabilitat condicional de la saturació donat el to assolint un nivell d'èxit del 37.3% i del 37.6% respectivament. Demostrant que el millor model de color a l'hora de classificar imatges és el CIELAB. Els errors de classificació entre els diversos descriptors no estan correlacionats per complet, i una millora notable en el rendiment s'obté mitjançant la fusió de totes les característiques abans esmentades aconseguint una efectivitat del 45.7% .

Segons explica l'article [1], també es va intentar determinar el percentatge d'encert de persones no capacitades en la tasca de classificació de imatges i descobrir si els percentatges d'encert aconseguits pels descriptors eren òptims al comparar-els amb els resultats obtinguts per éssers humans. Per tant van realitzar un estudi a través d'Amazon Mechanical Turk platform. A una serie de persones se'ls va mostrar una imatge de la base de dades i se'ls hi va dir que es va prendre en algun moment entre 1930 i 1980. Després se'ls va demanar que indiquessin a través d'opcions múltiples a quina dècada creien que va ser realitzada l'imatge (és a dir, tenien que

escollir entre “1930”, “1940”, “1950”, “1960” o “1970” ), no hi havia límit de temps imposat per prendre la decisió. Cadascuna de les 1.375 imatges a la base de dades es va avaluar per sis únics observadors, donant un total de 8,250 classificacions humanes. L'anàlisi de les respostes obtingudes va revelar que només un 26.0% de les fotografies van ser correctament classificades en la dècada que li pertocava. Cosa que només significa una petita millora respecte la probabilitat que s'obté quan les imatges es classifiquen l'atzar, que es del 20.0%. Per tant cal destacar que fins i tot els descriptors menys potents mostres individualment una millor precisió que un ésser humà que no ha estat entrenat. Apart, s'ha de destacar que el percentatge d'encert de tots els descriptors combinats és gairebé el doble que el d'un ésser humà no entrenat.

## Capítol 4:

# Nou descriptor

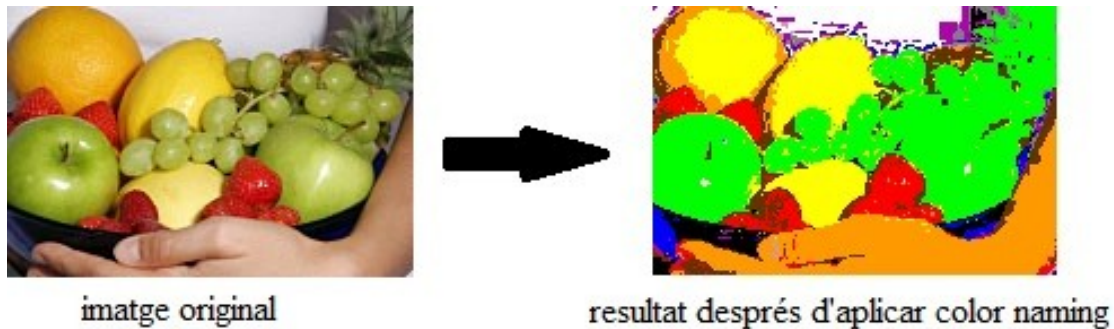
Com ja hem comentat anteriorment el nostre objectiu és millorar els resultats obtinguts per l'article comentat anteriorment. Per tal de poder portar a terme el nostre objectiu, varem creure interessant buscar un nou descriptor que avalués el color de les imatges. Al final ens varem decidir pel descriptor de Color Naming, desenvolupat per Benavente, Vanrell i Baldrich [11], el qual passem a explicar en el següent apartat.

### 4.1.Color Naming

Com indica l'article [11] el color naming és un model paramètric en el qual cada categoria de color està modelada com un conjunt difús amb funcions de pertinença paramètriques. Això a grans trets vol dir que donada una mostra d'entrada el model de color naming et retorna la probabilitat de pertànyer a cadascun dels 11 colors bàsics. A partir d'aquestes probabilitats, es pot extreure a quina categoria té més probabilitat de pertànyer la mostra. El nombre de categories que contempla el color naming és de 11, per tal d'escollir aquestes 11 categories s'han basat amb una serie d'estudis antropològics que demostren que que aquests 11 colors són els més identificatius per l'ésser humà. Els 11 noms de colors utilitzats en el descriptor són el vermell, el taronja, el marró, el groc, el verd, el blau, el violeta, el rosa, el negre, el gris i el blanc.

Explicat de manera més simple el que fa el descriptor és donat un píxel d'entrada et retorna una vector de 11 posicions, on en cada posició hi ha la probabilitat de que aquell píxel sigui aquell color. O sigui si per exemple tenim un píxel que el seu color esta entre el color verd i el blau en el vector de sortida tots els valors del vector tendiran a ser zero menys els del color verd i blau que tindran més probabilitat que sigui blau o verd. En la implementació del descriptor que hem utilitzat, li passem una imatge i el descriptor ens retorna un parell d'imatges i una matriu. La primera imatge que rebem és l'imatge original però pintant cada píxel amb el color més representatiu obtingut amb el color naming (figura 13), la segona imatge està etiquetada amb

una sèrie d'enters que representen el nom amb el major nombre de membres. I per últim, i el que ens interessa més, la matriu conté per cada píxel les probabilitats de que aquell píxel formi par de cada una de les 11 categories de color.



**Figura 13:** Exemple d'utilització del descriptor Color Naming

## 4.2.Procés d'adaptació del Color Naming

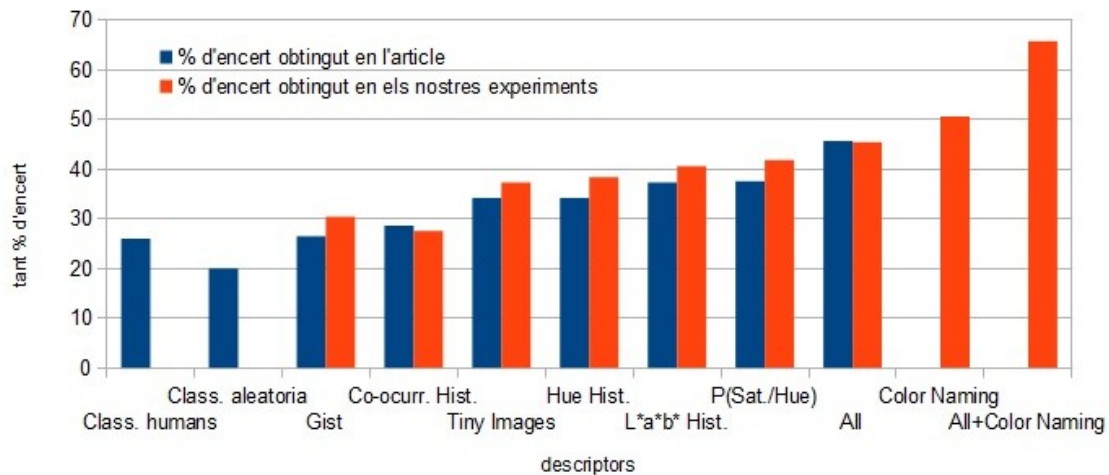
A l'hora d'intentar utilitzar la matriu amb el nostre experiment ens varem adonar que era complicat d'utilitzar ja que cada imatge tenia una mida diferent i per tant la matriu obtinguda també tenia una mida diferent, cosa que dificultava la seva utilització. Per això es va arribar a la conclusió que la millor solució seria fer l'histograma dels noms de colors que hi ha a la imatge. Aquest histograma el calculem fent la mitjana dels vectors de color naming de tots els píxels. De tal manera que ara per cada imatge passem a tenir un vector de 11 posicions on ens indica amb quina probabilitat quins són els colors que apareixen més en l'imatge, o sigui quins són els colors mes representatius i el menys representatius en l'imatge. Un cop tenim aquest nou vector ja podem repetir els experiments del article[1] un altre cop, per tal de comparar resultats.

## 4.3.Resultats

Un cop el nou descriptor va estar adaptat varem realitzar una series d'experiments per comprovar l'eficàcia del nou descriptor ja fós ell tot sol o juntament amb tots els descriptors. Els experiments realitzats van ser com els de l'article [1], utilitzant la mateixa base de dades i la mateixa quantitat de mostres per l'entrenament i el testeig. Els resultats que vam aconseguir



amb l'experiment del color naming tot sol van ser bastant sorprenents, com es pot comprovar en la figura 14, ja que es va aconseguir arribar a un 50.6% d'encert en el testeig de les imatges, un resultat molt per sobre dels aconseguits per tots els altres descriptors tot sols i, fins i tot, millorant els de tots els descriptors anteriors junts(45.7%).



**Figura 14:** Comparació amb els resultats explicats en l'apartat 3.4 i els resultats obtinguts per el Color Naming i tots els descriptors més el Color Naming

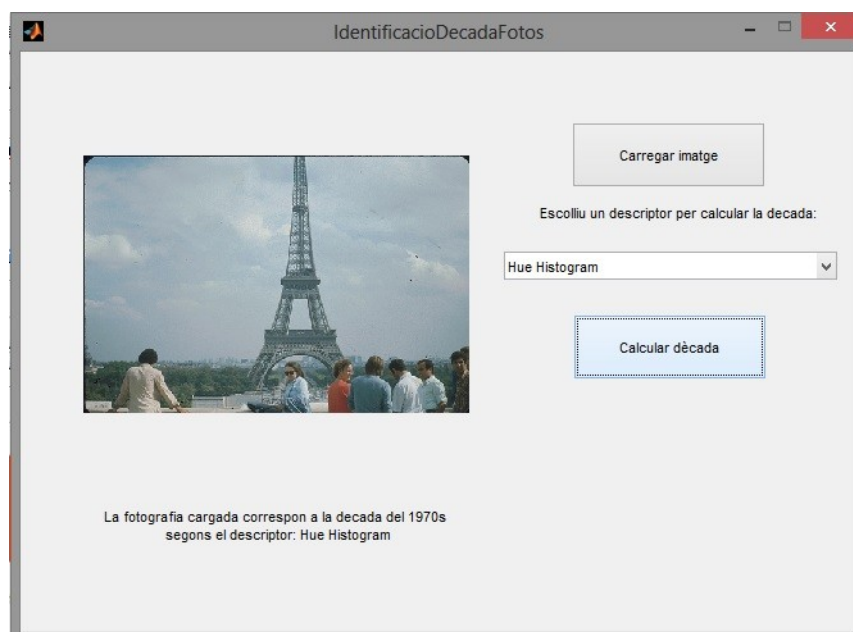
Aquests resultats ens van portar a la conclusió de que molt possiblement cada dècada té com un color més utilitzat, ja sigui pels colors que apareixen o per el tipus de color que les imatges adquirien amb la càmera representativa de la dècada. Un cop vist els resultats del color naming tot sol, vam testear els resultats que s'obtenien si s'unien tots els descriptors junts incloent el color naming. Els resultats que varem obtenir també van ser molt bons, fent que el percentatge d'encert s'elevés fins al 65.84%, implicant un altre cop una gran millora respecte els altres experiments. Per tant creiem que el descriptor color naming és una molt bona eina per tal de classificar imatges i que s'ha de tenir en compte en futures implementacions.

## Capítol 5:

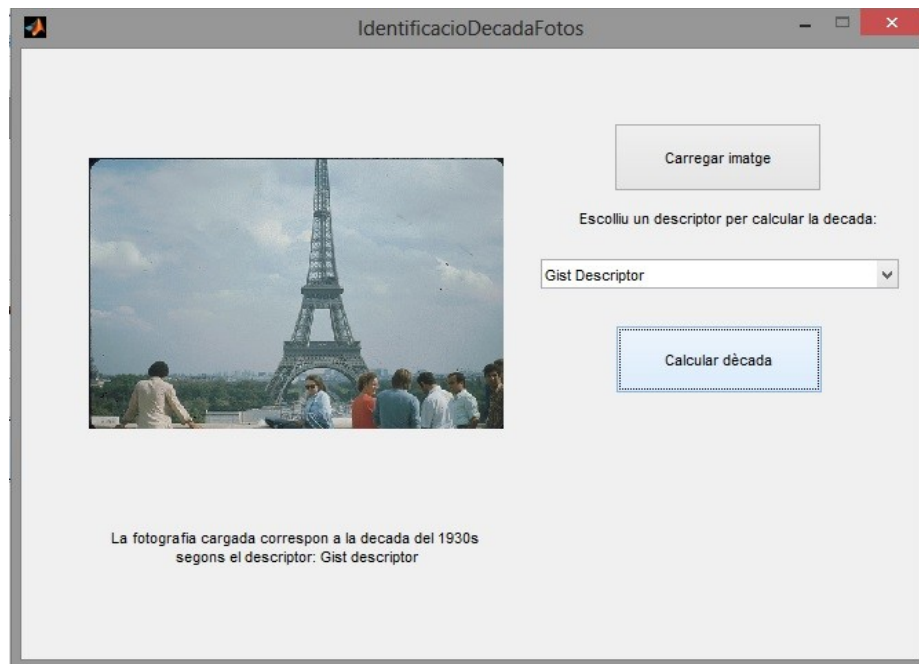
# Interfície gràfica per classificar imatges

En informàtica, una interfície gràfica d'usuari (també anomenada GUI) és un tipus d'interfície d'usuari que permet als usuaris interactuar amb els dispositius electrònics usant imatges en lloc d'ordres de text. Les interfícies gràfiques d'usuari es poden utilitzar en ordinadors, dispositius portàtils com reproductors de MP3, electrodomèstics, màquines industrials... Una interfície gràfica d'usuari representa la informació i les accions disponibles per a un usuari a través d'icones gràfics i indicadors visuals, en contraposició a les interfícies basades en text. Les accions es realitzen generalment a través de la manipulació directa dels elements gràfics.

En el nostre cas per tal de crear la GUI per el classificador d'imatges ens varem decantar per escollir l'editor d'interfícies gràfiques de Matlab (també anomenat GUIDE). Varem escollir realitzar la interfície amb el GUIDE de Matlab perquè la gran majoria del codi que utilitzem per realitzar la classificació és de Matlab, per tant per nosaltres és molt més senzill adaptar el codi a la interfície gràfica. Apart d'això, GUIDE té totes les característiques bàsiques de tots els programes visual com Visual Basic o Visual C++.

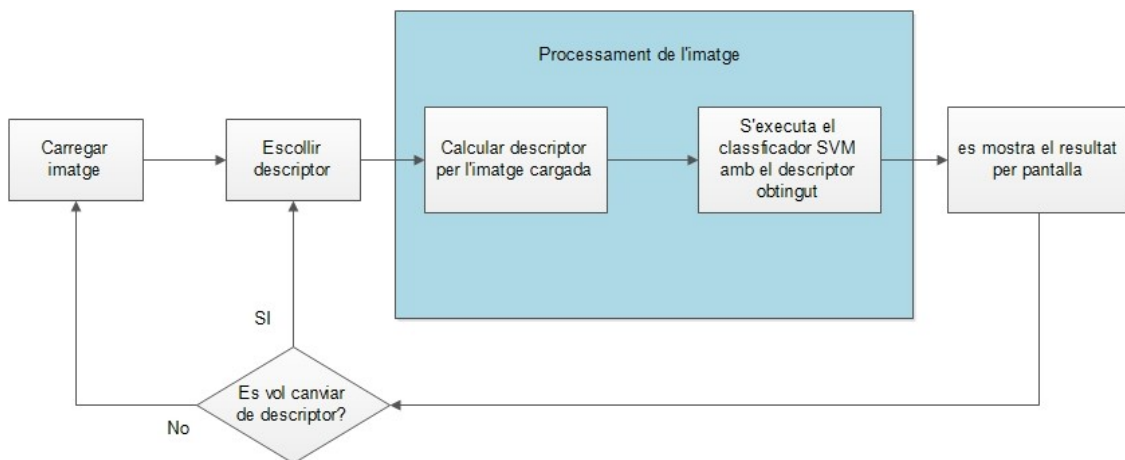


**Figura 15:** Exemple d'una bona classificació de l'aplicació utilitzant el descriptor hue histogram



**Figura 16:** Exemple d'una mala classificació de l'aplicació utilitzant el descriptor gist

La interfície que hem realitzat és molt senzilla i fàcil de utilitzar com es pot veure en la figura 15, per tal que qualsevol persona encara que tingui un nivell baix d'informàtica no trobi cap dificultat a l'hora d'utilitzar-la. Un cop l'usuari ha obert la interfície només te tres opcions, la primera seria carregar una imatge en l'aplicació, la segona seria una llista desplegable que serveix per escollir amb quin descriptor es vol classificar l'imatge, i la tercera opció seria la de calcular dècada que com el seu nom indica és el botó que executa tot el codi per tal de classificar l'imatge. Un cop es prem el botó de calcular dècada, l'aplicació primerament calcula els descriptors de l'imatge i els guarda a disc. Un cop els descriptors estan calculats, l'aplicació comença l'execució del classificador SVM segons el descriptor escollit i al cap de pocs segons es mostra el resultat en la interfície. Com es pot comprovar en les figures 15 i 16 una mateixa imatge es classificada amb un diferent resultat depenent del descriptor utilitzat, en aquest cas l'imatge es dels anys 70 i el descriptor hue histogram la classificat correctament, mentres que el descriptor gist la classificada en els anys 30. En el següent diagrama es pot veure tot el cicle que segueix l'aplicació.



**Figura 17:** Diagrama que explica l'execució que segueix l'aplicació creada en aquest projecte

En la creació d'aquesta aplicació el codi base que hem utilitzat ha estat el de l'article[1], encara que hi hem tingut de fer unes modificacions per tal de que complís amb el nostre objectiu. Per tal de poder executar el nou descriptor varem haver d'afegir tot el codi necessari per tal de calcular el descriptor color naming.

En el cas del codi encarregat d'executar el classificador SVM és el que ha sofert més canvis. Ja que el codi del article [1] estava pensat per a entrenar i testear bases de dades que ja estan classificades, o sigui que només esta pensat per fer experiments, en canvi el nostre codi només ha de testear una imatge i pot prescindir de l'entrenament del classificador. Per tant el que hem fet ha estat modificar el codi del classificador per tal que només accepti una imatge. També hem suprimit tota la part d'entrenament del classificador, però per tal de poder executar el codi em guardat el models d'entrenament per cada descriptor, prèviament calculats amb el codi del article[1]. El fet de no haver de realitzar l'entrenament també ens reporta beneficis en temes d'execució temporal i de memòria, ja que l'execució de l'entrenament és un procediment bastant lent i els descriptors necessaris per fer l'entrenament ocupen molta memòria.

## Capítol 6:

# Conclusions i possibles millores

### 6.1.Conclusions

La idea d'aquest projecte va sorgir de la necessitat que hi ha en classificar grans volums d'imatges ja sigui per dècades, estils o qualsevol altra possible classificació. Bàsicament el primer objectiu consistia en facilitar la vida als usuaris a la classificació d'imatges. Quan varem trobar el article [1] i varem veure que feien exactament el que estàvem pensant, varem creure que seria interessant millorar els resultats que ells obtenien i adaptar el seu estudi per tal que el poguessin utilitzar les persones que tenien un nivell més baix d'informàtica.

Com a objectiu principal ens varem proposar la de millorar els resultats obtinguts per l'article[], fet que hem aconseguit amb el descriptor de color naming. Encara que els resultats continuen estan lluny del percentatge d'èxit desitjats creiem que estan força bé i son bastants positius. Ens hagués agradat aconseguir algun altre descriptor per fer més proves però no ha estat possible per raons de temps.

En el cas de facilitar la classificació d'imatges a qualsevol usuari, creiem que ho hem aconseguit amb l'aplicació que hem realitzat que és molt fàcil de fer servir i no presenta cap complicació. En general creiem que el projecte ha estat positiu, però que necessitaria ser millorat perquè sigués encara més interessant. Per això a continuació proposem algunes possibles millores que podrien fer més atractiu el projecte.

### 6.2.Possibles millores

Lògicament una possible millora seria intentar aconseguir amb nous descriptors uns millor resultats a l'hora de classificar les imatges. Encara que els resultats obtinguts després d'afegir el descriptor color naming al codi són bastant bons, creiem que seria factible trobar algun altre descriptor que almenys ajuntat amb tots els altres aconseguís millorar el percentatge

d'efectivitat. L'objectiu seria arribar a un percentatge pròxim al 75% fet que faria que el classificador fos molt fiable.

Un cop s'hagués millorat l'èxit del classificador, seria interessant també donar l'opció a l'usuari de poder classificar grans volums d'imatges de manera automàtica. O sigui, que l'usuari tingués l'opció de seleccionar una carpeta i l'aplicació s'encarregues de classificar totes les imatges, creant carpetes per cada dècada i col·locant les imatges a la seva carpeta corresponent. Però això només seria factible i interessant si el percentatge d'encert del classificador fós molt elevat. També es podria incrementar el nombre de dècades classificades fins a la dècada dels noranta. Ja que a partir del any 2000, ja es va fer molt extens el ús de les càmeres digitals i en alguns aspectes les fotografies digitals podrien complicar la classificació.

L'última millora que proposaríem seria la de utilitzar aquesta aplicació en fer algun altre tipus de classificació. Per exemple també es podria utilitzar per classificar quadres d'art dintre els diferents corrents artístics que han existit al llarg de l'història. Això en principi creiem que seria bastant senzill ja que només s'hauria de modificar petits paràmetres en el codi. El problema més gran seria trobar o construir una base de dades prou extensa i ben classificada que ens servis per aconseguir els diferents models d'entrenament.

# Bibliografia

- [1] - F. Palermo, J. Hays, A. A. Efros. Dating Historical Color Images. European Conference on Computer Vision, LNCS, vol. 7577, pp. 499-512, 2012.
- [2] - Y. Luo, X. Tang. Photo and Video Quality Evaluation: Focusing on the Subject. European Conference on Computer Vision, LNCS, vol. 5304, pp.386-399, 2008.
- [3] - P. Isola, J. Xiao, A. Torralba, A. Oliva. What makes an image memorable?. Conference on Computer Vision and Pattern Recognition, pp. 145-152, 2011.< an lang=CA>
- [4] - D. Cohen-Or, O. Sorkine, R. Gal, T. Leyvand, Ying-Qing Xu. Color Harmonization. ACM Siggraph 2006, pp.624-630, 2009.
- [5] – Francesc J. Ferri, Universitat de València, Apunts de l'assignatura de visió per computador (2008): <http://informatica.uv.es/iigua/VC/VC-1intro-ho.pdf>
- [6] – Wikipedia, Computer vision [http://en.wikipedia.org/wiki/Computer\\_vision](http://en.wikipedia.org/wiki/Computer_vision)
- [7] - Qiang Ji, Rensselaer Polytechnic Institute, New York, apunts: [http://www.ecse.rpi.edu/Homepages/qji/CV/3dvision\\_intro.pdf](http://www.ecse.rpi.edu/Homepages/qji/CV/3dvision_intro.pdf)
- [8] – Wikipedia, aprenentatge automàtic: [http://ca.wikipedia.org/wiki/Aprenentatge\\_automàtic](http://ca.wikipedia.org/wiki/Aprenentatge_automàtic)
- [9] -Wikipedia, Màquina de vector de suport, en català i anglès: [https://ca.wikipedia.org/wiki/Màquina\\_de\\_vector\\_de\\_suport](https://ca.wikipedia.org/wiki/Màquina_de_vector_de_suport) i [https://en.wikipedia.org/wiki/Support\\_vector\\_machine](https://en.wikipedia.org/wiki/Support_vector_machine)
- [10]- Cristina Martínez Ruedas, Universitat de Sevilla, detector multiusuario para DS-CDMA basado en SVM,(2006) : [http://bibing.us.es/proyectos/abreproy/11185/fichero/Volumen+1\\_Detector+Multiusuario+para+DS-CDMA+basado+en+SVM%252F7.+Support+Vector+Machines%252FSupport+Vector+Machines.pdf](http://bibing.us.es/proyectos/abreproy/11185/fichero/Volumen+1_Detector+Multiusuario+para+DS-CDMA+basado+en+SVM%252F7.+Support+Vector+Machines%252FSupport+Vector+Machines.pdf)
- [11] – R.Benavente, M.Vanrell, R.Baldrich. Parametric fuzzy sets for automatic color naming. Journal of the Optical Society of America A, vol.25, num. 10, pp. 2582-2593, 2008.

## **5231 - APLICACIÓ PER A LA PREDICCIÓ AUTOMÀTICA DE LA DATA D'ADQUISICIÓ D'UNA IMATGE**

### **Resum**

En aquesta memòria es presenta un projecte que té com a objectiu principal la creació d'una aplicació que, donada una imatge d'entrada, retorni en quina dècada va ser adquirida la imatge. Apart es pretén millorar els resultats obtinguts en altres estudis sobre la classificació d'imatges segons la seva data d'adquisició. En la memòria s'explica l'estudi realitzat anteriorment sobre el tema i quin ha estat el mètode escollit per millorar els seus resultats. També s'explica com hem creat l'aplicació i els passos que segueix l'aplicació en la seva execució.

### **Resumen**

En esta memoria se presenta un proyecto que tiene como a objetivo principal la creación de una aplicación que, dada una imagen de entrada, devuelva en qué década fue adquirida la imagen. Aparte se pretende mejorar los resultados obtenidos en otro estudio sobre la clasificación de imágenes según su fecha de adquisición. En la memoria se explica el estudio realizado anteriormente sobre el tema y cuál ha sido el método escogido para mejorar los resultados. También se explica como hemos creado la aplicación i los pasos que sigue la aplicación en su ejecución.

### **Summary**

In this project, we present a project that has the goal of developing an application to predict the decade in which an image was acquired. Another aim of the project is to improve the results of a previous study about image classification in terms of the acquisition date of images. The report covers the previous study on this subject and the method that has been chosen to improve the performance of the previous work. The report also explains how the application has been created and the steps that this application follows when it is executed.